UNSUPERVISED MONOCULAR DEPTH ESTIMATION: LEARNING TO GENERALIZE

BY

DANIEL GONZALES

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2020

Urbana, Illinois

Adviser:

Professor Minh Do

ABSTRACT

Models for unsupervised monocular depth estimation (MDE) have gained much attention due to recent breakthroughs and the ability to train with unlabeled data. Despite the state-of-the-art methods performing well on depth prediction benchmarks, certain artifacts and their performance compared to their supervised counterparts make them less favorable in certain domains. This thesis analyzes these models and presents a set of methods for improvement which can be applied in the training process.

Recent papers in unsupervised MDE focus on increasing performance metrics on the KITTI benchmark. We show that the results from these methods can be further improved by (i) providing synthetic training data via the game engine Grand Theft Auto V (GTAV) and (ii) applying data augmentation techniques that are consistent with the camera intrinsic parameters of the model.

To my parents, Phraes, and Swims for their love and support.

ACKNOWLEDGMENTS

I first thank Tian Ma and Sandia National Laboratories for their help and contributions. This work is partly supported by them through the Lab Directed Research and Development (LDRD) grant.

I gratefully acknowledge my adviser, Prof. Minh Do. His wisdom and enthusiasm has not only guided me in the completion of this work, but is the main reason I found passion in the fields of signal processing and vision. I also thank the rest of my colleagues of the Computational Imaging Group. I learned so much from them all and their support has helped me accomplish so much during my time in graduate school.

I thank my mom and dad, Claudia and Eric Gonzales, for all their love that has kept me strong and well. I am grateful for their encouragement during my time at UIUC which has given me so many opportunities, shaping me into the person I am today. I'm also grateful to my sister Phraes and her companion Swimmy, for their undying moral support and strange sense of humor.

I lastly thank my closest friends. They have been there for me with their support, jokes, and insight. We have seen each other grow throughout the years and I cherish their friendship deeply: Ekant Desai, George Magallanes, Austin Stanton, Tony & Brent Wu, Matthew Walker, Gabriel Malca, Rajeev Udumula, Drew Beeman, and Cynthia & Chen Kachanthong.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Monocular depth estimation (MDE) is the prediction of depth for every pixel in a single RGB image. This can be challenging, especially since traditional computer vision algorithms require a second image taken from another perspective to calculate depth. However, humans can intuitively estimate 3D structure from only one picture due to their past experiences interacting with the world. Recent state-of-the-art MDE models are coming closer to emulate this.

In a supervised setting, MDE models are trained using ground-truth depth data. However, this can be difficult to acquire due to expensive depth sensors, some of which supply only sparse data. Unsupervised MDE circumvents the issue by using structure from motion (SfM) techniques: giving the ability to learn structure without depth data. This convenience in data collection, along with recent breakthroughs [1], has given unsupervised MDE a lot of attention in the research community. As these methods matured, their performance metrics have become more comparable to their supervised counterparts. The ease of data collection also allows wider use in applications such as self-driving vehicles, augmented reality, and robotic depth perception.

Despite getting astounding results on depth prediction benchmarks such as KITTI, unsupervised MDE methods face their own set of challenges. Certain objects are unable to be predicted with reliable structure, and in some cases the model may completely ignore the depth of an object on the road. These are the kind of problems that keep unsupervised MDE from being deployed in certain domains. The content of this thesis is separated into three contributions: (1) an analysis on unsupervised MDE methods and their generalizability, (2) a photo-realistic synthetic dataset used to increase performance in the self-driving domain, and (3) a data augmentation training procedure that is consistent with the loss function of these models. We show through our experiments how these techniques can help increase the performance and

reliability of unsupervised MDE.

CHAPTER 2

RELATED WORK

2.1 Unsupervised MDE Methods

Unsupervised or self-supervised depth estimation is learned by using 2D image sequences that consist of camera motion. Once learned, the model can predict depth using only an RGB image. This is typically done using a sequence of stereo image pairs or monocular video, the latter being the focus of this thesis. Training with monocular sequences has added ambiguity in structure and scaling, but data is easier to obtain. Both methods have generally the same procedure starting with [1]: two networks for pose and depth are used. The Pose CNN learns the ego-motion between frames of the sequence which is used to constrain the learning of Depth CNN, effectively decreasing the ambiguity of 2D inputs.

Since these methods are based on SfM, assumptions on the training data are expected for proper 3D-reconstruction, e.g. camera motion, static scenes, and no occlusions or reflective surface. Different approaches apply their own novelties to help mitigate assumption-breaking data from contaminating the learning process. For instance, per-pixel masks are used to distinguish between valid and assumption-breaking pixels. Zhou et al. [1] do this using weighted "explainability" values for the mask whereas later methods ([2], [3], [4]) use binary masks. Some approaches add further constraints to enforce consistency with other signals such as optical flow, edges, and surface normals ([3], [5]). The method in [4] addresses occlusions and disocclusions by reformulating the projection loss function. All these novel additions have increased the performance of unsupervised MDE, but the state-of-the-art still faces failure cases from assumption-breaking data, resulting in artifacts such as infinite-depth holes for certain objects and poorly estimated structure of complicated shapes or highly reflective surfaces [4].

2.2 Synthetic Data

In cases where it is difficult to acquire, generating data synthetically may be a viable option for increasing model performance by means of supplementation. For MDE, this may be more useful in some domains than others e.g. autonomous driving ([6], [7], [8]). Models can benefit from simulation frameworks that are highly interactable and have photo-realistic environments so the generated data is as close to the target data's distribution as possible.

Simulated data may also come with additional insight unavailable in the authentic data. The authors of [9] take advantage of the ground-truth dense depth of synthetic data to increase performance on KITTI, a dataset with only sparse ground-truth depth data. This approach was done in combination with image style transfer in order to adapt to the domain of KITTI. Note that this approach was not attempted on unsupervised MDE methods, and may likely give poorer results due to the inconsistency between adjacent frames when style transfer is applied.

2.3 Data Augmentation for MDE

To increase the generalization performance of a model, transforming the input training data is a common practice. However, data augmentation policies in MDE are very simple compared to other deep learning tasks. Since affine transformations artificially alter the camera intrinsic parameters, unsupervised approaches stay away from them to avoid affecting the reprojection loss. In the case of supervised MDE, affine transformations such as rotations cause invalid pixel data for the corresponding ground-truth depth [10]. The authors of [11] introduce augmenting data via style randomization resulting in performance increases when trained on synthetic data. However, this policy was not assessed when using authentic training data. In the end, most MDE augmentation policies consist of only horizontal flipping and photometric transformations.

CHAPTER 3

ANALYSIS ON UNSUPERVISED MDE

We begin by assessing the qualitative results of an unsupervised MDE network for generalization performance. This is done using the model in [4] trained on KITTI 2015 Eigen split. We then use it to analyze the depth network's output when feeding in scenes with injected objects. We vary these objects in size, location, and object-type in handpicked scenes that represent the KITTI data. Objects chosen are either cropped from KITTI validation images or from external sources. We measure depth of objects by averaging over estimated depth values that overlap with the object segmentation mask.

3.1 Object Size

We expect certain objects such as stop signs, cars, and soccer balls to each have a general size given our past experiences with them. So when we see these familiar objects in images, we instinctively estimate depth by using them for scale: an image of a scene with a stop sign that takes up few pixels

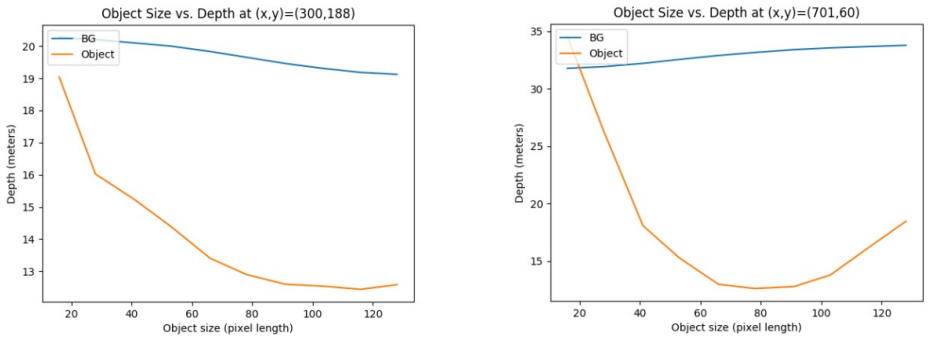

Figure 3.1: Typical plots of object depth vs. pixel size. We expect an inverse relationship between the two variables (left). However, sometimes the relationship breaks down with large object pixel sizes (right).

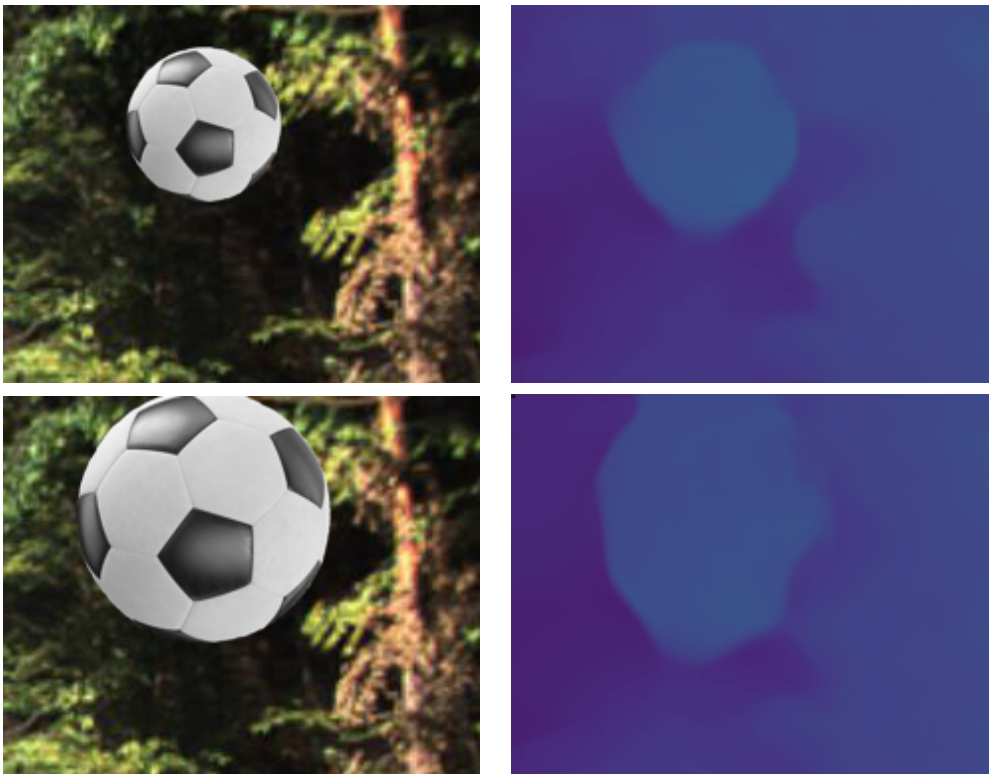

Figure 3.2: Depth maps of injected objects at different pixel sizes. For pixel size 64 (top), the depth edges are closer to actual object structure than size 128 (bottom).

is a cue that the sign is probably very far. There are obviously exceptions to this. The sign may be a miniature or there may be conflicting depth cues from other objects. However, we expect a relationship between depth and object size in pixels, assuming the object size in world-coordinates is fixed. Based on the pinhole camera projection model from [12], the projection equation (3.1) shows a point (x, y, z) in world-coordinates is inversely related by depth z and scaled by focal length f, mapping to pixel-coordinate (u, v).

$$(x, y, z) \rightarrow (u, v) = (f\frac{x}{z}, f\frac{y}{z}) \tag{3.1}$$

This experiment assesses if this relationship holds under the model's estimated depth outputs. Object pixel size is varied from 16 to 128 pixels in side length. Besides a few exception cases which we describe in Section 3.2, the model is able to roughly estimate the inverse mapping for certain object pixel size ranges. Figure 3.1 contains the typical graphs across scenes and object types. The first behavior is an expected relationship between the two variables. However, the second behavior shows a trend of divergence as object pixel size becomes too large. Figure 3.2 shows an example of this, and it is shown that the estimated depth structure breaks down.

We reason that this behavior is due to the global limitations of the CNN that the model's depth network is based on. Because the receptive field of these networks only cover a segment of the input features at a time, depth of an object can be harder to estimate if it spans the entire field. When also considering the average pixel size of objects in the training data, these models may degrade in performance when objects get too close to the camera.

3.2 Object Location

Following the argument of object size in Section 3.1, we expect that an object of fixed pixel location and pixel size will have roughly the same estimated depth in different scenes. Similarly, small shifts in pixel location should not drastically affect depth. However, experiments show this is not the case. Depth of the object is heavily dependent on the neighborhood that surrounds it.

Experiments were ran by scanning objects with fixed pixel size across dif-

Figure 3.3: Examples of estimated object depth of Unsupervised MDE. Depth of injected objects are heavily influenced by the ground plane they perceivably rest on.

ferent scenes and recording their depth. Results of the model show three different general behaviors. The first behavior is present when the object overlaps with the ground of the scene. Figure 3.3 shows that the depth is heavily dependent on the plane that it perceivably rests on. This suggests that the model learns depth of objects based on consistency with surroundings rather than scale of the object type.

Figure 3.4 shows the behavior in the case where the injected object overlaps other objects in the scene. The model now estimates the depth of the object as an extension of the object it overlaps. This is not expected behavior, since we expect the object's depth to be estimated as in front of the object it overlaps. This could suggest that the model infers structure based on edges between background and foreground patterns.

The last behavior is presented in Figure 3.5. It is most common in scenes where the car is driving down a straight road. Objects that are overlap the road at the bottom of the image are not recognized by the model. Instead, the depth of the object is estimated as the rest of the road. Figure 3.6 shows the object depth matches the background depth until we shift the object off the road, and the structure is estimated again. We also notice similar behavior whent trained on our synthetic data discussed in Chapter 4. This depth "blindspot" may be a result of overfitting on the training data. Objects on the road at this location are rare to see in KITTI and are usually dynamic, meaning they will have little contribution to the loss function. Artifacts such

Figure 3.4: Examples of estimated object depth of Unsupervised MDE. Depth of injected objects are estimated as extensions of structure to scene objects.

as this affect the practicality of these models. They may also not be properly assessed by error metrics since the ground-truth pixels are concentrated in the middle rows of each image.

3.3 Object Type

We experimented with different object types to inject into the scene. This includes vehicles that were cropped from KITTI images as well as basic shapes like circles. We noticed the model was able to generalize some depth of injected foreign objects. Instances such as solid-colored circles and soccer balls had crisp depth edges.

When injecting vehicle objects into the scene, similar behavior was observed. However, the structures of the vehicles were more three-dimensional than foreign objects like the soccer ball, which was estimated as more flat. The model behaved worse on objects with more complicated shapes. Injected objects such as signs resulted in depth edges that were simplified or were not recognized. Estimating depth of these kinds of shapes is a known problem for these models [4].

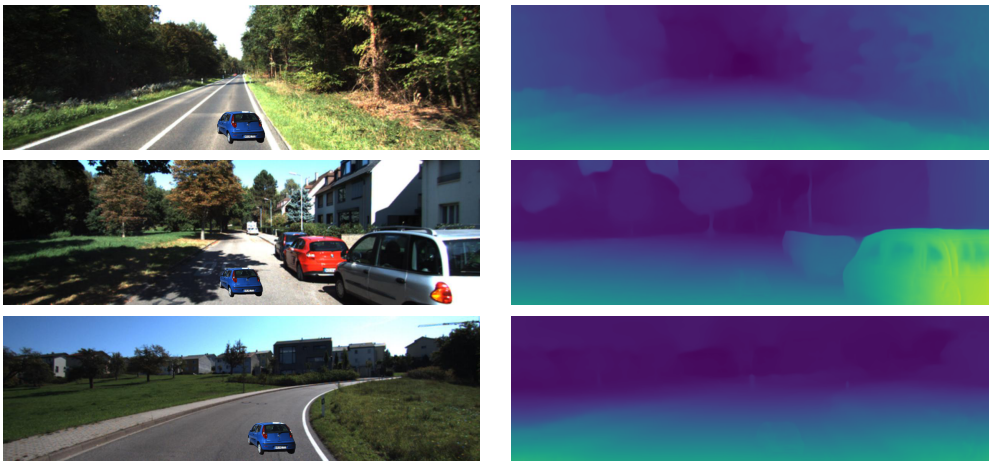

Figure 3.5: Examples of estimated object depth of Unsupervised MDE. Depth of injected objects are estimated as the depth of the road, ignoring most structure of the object.

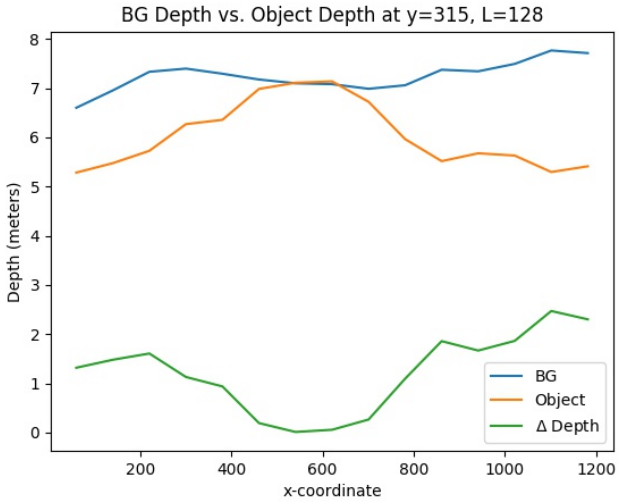

Figure 3.6: Estimated depth of object scanned horizontally over a depth blindspot. The object is scanned across the bottom of input image.

CHAPTER 4

SYNTHETIC DATA FOR UNSUPERVISED MDE

Although data collection is easier for unsupervised MDE methods, applications such as self-driving may still be difficult. This is because the nature of the data breaks many assumptions: highly dynamic scenes with many occlusions and reflective surfaces. Self-driving models also need data in a wide variety of environments in order to generalize well. Using synthetic data can help in these situations.

We use the GTAV simulator by [6] and refined by [7] to develop a synthetic dataset that is supplemented to KITTI training data with the goal of increasing performance in unsupervised MDE. This simulator was chosen for its similarities to KITTI data and high configurability. Our dataset contains frames captured at 20 Hz for higher temporal granularity. Frames were processed to have the same resolution (1242x375) and camera intrinsic parameters. The camera orientation and vehicle were chosen to resemble the target configuration as close as possible. The dataset is composed of over 88,000 frames taken in environments that are similar to the target domain. Samples are shown in Fig. 4.1. Data from the residential areas and over half of the open road areas are static scenes. Inspired by [13], we force all vehicles and other objects to be frozen in place, resulting in the training data containing more valid pixels that obey the model assumptions. Lastly, we include extra data that is not found in the target data, including dense depth maps, camera extrinsic parameters, and vehicle segmentations.

Figure 4.1: Comparison samples between synthetic GTAV data (left) and target KITTI data (right).

CHAPTER 5

LOSS-CONSISTENT DATA AUGMENTATION

Augmentation policies for unsupervised MDE are relatively simple, consisting of only horizontal flipping and photometric transformations (hue, saturation, channel-flipping, etc.). Other affine transformations are avoided due to the nature of calculating the loss function. For [4] and other unsupervised MDE methods like it, the photometric reprojection error L_p is minimized for each image frame I_t and its adjacent frames $I_{t'}$ where $t' \in \{t-1, t+1\}$. The projective transform is then taken for frames $I_{t'}$ using the predicted depth map D_t, ego-motion $T_{t\to t'}$, and intrinsics K, resulting in an aligned image $I_{t'\to t}$ to original image I_t. This is a pixel-wise transformation for all p and is shown below:

$$L_p = \sum_{t'} \sum_{p} pe\big(I_t(p), I_{t'\to t}(p)\big) \tag{5.1}$$

where pe is the photometric error function justified in [14], [15]. The following projective transform maps homogeneous pixel coordinates p in $I_{t'}$ to p' in $I_{t'\to t}$ with some scaling factor:

$$p' \sim K T_{t\to t'} \big(D_t(p) \cdot K^{-1} p\big) \tag{5.2}$$

In order for an affine transform A to be consistent with L_p, it should be applied to I_t and its adjacent frames $I_{t'}$. We define the transformed I_t as $\hat{I}_t = AI_t$ and similarly for $\hat{I}_{t'}$. This results in pe to take $\hat{I}_t$ and $\hat{I}_{t'}$ as input as well as the depth and pose networks, outputting $\hat{D}_t$ and $\hat{T}_{t\to t'}$.

CHAPTER 6

EXPERIMENTS

We assess the quality of our contributions through different studies. Specifically, we experiment with the synthetic GTAV dataset and data augmentation policies to see how they can increase results of unsupervised MDE methods and their performance on the KITTI 2015 dataset. These experiments are tested using the Monodepth2 model by [4].

6.1 Synthetic Data Pretraining

Unsupervised MDE models have seen improvement in performance when pretrained on certain datasets [4]. This experiment takes the idea further by utilizing synthetic data similar to the target domain. More specifically, we use a model pretrained on our synthetic data to substitute the requirement of additional training data from the target domain. This allows us to get similar performance using less target domain data by taking advantage of synthetic data pretraining. Table 6.1 shows the results of our method. With only half the data, we are able to achieve similar metrics to the baseline. At ten percent, the model is still able to give comparable performance.

Table 6.1: Results of synthetic data pretraining tested on KITTI 2015 Eigen split. Different percentages of training data are kept in the training set with samples selected at random. The baseline uses no pretraining and is trained on the original Eigen split.

Train	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
baseline	0.125	0.983	5.035	0.205	0.859	0.954	0.979
100% data	0.129	1.098	5.268	0.210	0.855	0.951	0.977
50% data	0.131	1.071	5.166	0.211	0.849	0.950	0.978
25% data	0.138	1.199	5.293	0.218	0.841	0.946	0.976
10% data	0.140	1.165	5.322	0.219	0.833	0.945	0.976

Table 6.2: Results of unsupervised MDE model with different affine transform data augmentation policies: (S) - Scale, (R) - Rotation, (C) - Crop.

Train	Abs Rel	Sq Rel	RMSE	RMSE log	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
baseline	0.125	0.983	5.035	0.205	0.859	0.954	0.979
S	0.126	1.056	5.116	0.207	0.860	0.953	0.978
R	0.128	1.064	5.146	0.209	0.856	0.951	0.978
S + C	0.131	1.108	5.265	0.213	0.852	0.949	0.977
S + C + R	0.132	1.138	5.217	0.212	0.854	0.950	0.977

We have tried other configurations with synthetic data such as joint training with target domain data. This however did not lead to improvement as we believe the distribution of each domain is still too different.

6.2 Data Augmentation Policy Study

In this experiment, we add different affine transforms into the data augmentation policy of an unsupervised MDE model in attempt to increase performance metrics. We train a model using the following affine transforms: rotation, resizing, and cropping. Each affine transform is performed independently with 0.5 probability. Table 6.2 shows the results of different data augmentation policies.

The results show little to no improvement when adding affine transforms. However, we have noticed smoother loss curves, and more stable training. Further experimentation in future work may point to better generalization performance.

CHAPTER 7

CONCLUSION

This thesis has illustrated ways for unsupervised MDE models to learn how to generalize better in order to close the performance gap between supervised counterparts. Our work has given insight on how unsupervised MDE models estimate the depth of objects in a scene, and shows areas where these models can improve. Our other two contributions are methods that can be applied to these models in order to increase generalization performance. This includes utilizing synthetic data that better follows model assumptions and that closely resembles the target data. We also show how adding a loss-consistent data augmentation policy can maintain the integrity of data during training. Together, these contributions are able to improve the performance of unsupervised MDE models both qualitatively and quantitatively.

The methods of this thesis also leave room for future work. For instance, the global limitations of MDE models shown in Section 3.1 give direction to more spatial-aware approaches such as [16]. We can also experiment with the use of image/video style transfer from methods like [17], [9] in order to adapt synthetic data to align more closely with the distribution of the target data. Other forms of transformations (projective) and policies for data augmentation can be further explored to find a policy that is more effective for learning MDE. Also, by having access to the camera extrinsic parameters of the synthetic data, we can better evaluate the pose network of unsupervised MDE models to find more efficient ego-motion constraints when learning depth.

REFERENCES

[1] T. Zhou, M. Brown, N. Snavely, and D. G. Lowe, "Unsupervised learning of depth and ego-motion from video," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul 2017. [Online]. Available: http://dx.doi.org/10.1109/cvpr.2017.700

[2] R. Mahjourian, M. Wicke, and A. Angelova, "Unsupervised learning of depth and ego-motion from monocular video using 3d geometric constraints," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5667–5675.

[3] Z. Yang, P. Wang, Y. Wang, W. Xu, and R. Nevatia, "Every pixel counts: Unsupervised geometry learning with holistic 3d motion understanding," in *Computer Vision – ECCV 2018 Workshops*, L. Leal-Taixé and S. Roth, Eds. Cham: Springer International Publishing, 2019, pp. 691–709.

[4] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, "Digging into self-supervised monocular depth estimation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 3828–3838.

[5] Z. Yang, P. Wang, Y. Wang, W. Xu, and R. Nevatia, "Lego: Learning edge with geometry all at once by watching videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 225–234.

[6] A. Ruano Miralles, "An open-source development environment for self-driving vehicles," M.S. thesis, Universitat Oberta de Catalunya, May 2017.

[7] B. Hurl, K. Czarnecki, and S. L. Waslander, "Precise synthetic image and lidar (PreSIL) dataset for autonomous vehicle perception," *CoRR*, vol. abs/1905.00160, 2019. [Online]. Available: http://arxiv.org/abs/1905.00160

[8] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," *arXiv preprint arXiv:1711.03938*, 2017.

[9] A. Atapour-Abarghouei and T. P. Breckon, "Real-time monocular depth estimation using synthetic data with domain adaptation via image style transfer," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2800–2810.

[10] I. Alhashim and P. Wonka, "High quality monocular depth estimation via transfer learning," *arXiv preprint arXiv:1812.11941*, 2018.

[11] P. T. Jackson, A. Atapour-Abarghouei, S. Bonner, T. Breckon, and B. Obara, "Style augmentation: Data augmentation via style randomization," *arXiv preprint arXiv:1809.05375*, pp. 1–13, 2018.

[12] D. A. Forsyth and J. Ponce, *Computer Vision - A Modern Approach, Second Edition.* Pitman, 2012.

[13] Z. Li, T. Dekel, F. Cole, R. Tucker, N. Snavely, C. Liu, and W. T. Freeman, "Learning the depths of moving people by watching frozen people," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[14] H. Zhao, O. Gallo, I. Frosio, and J. Kautz, "Loss functions for image restoration with neural networks," *IEEE Transactions on Computational Imaging*, vol. 3, no. 1, pp. 47–57, 2017.

[15] C. Godard, O. Mac Aodha, and G. J. Brostow, "Unsupervised monocular depth estimation with left-right consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 270–279.

[16] R. Liu, J. Lehman, P. Molino, F. Petroski Such, E. Frank, A. Sergeev, and J. Yosinski, "An intriguing failing of convolutional neural networks and the coordconv solution," in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Curran Associates, Inc., 2018, pp. 9605–9616. [Online]. Available: http://papers.nips.cc/paper/8169-an-intriguing-failing-of-convolutional-neural-networks-and-the-coordconv-solution.pdf

[17] J. Yoo, Y. Uh, S. Chun, B. Kang, and J.-W. Ha, "Photorealistic style transfer via wavelet transforms," in *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.